Neural Network Architectures for E-commerce Product Categorization

Abstract

This report presents the implementation of three neural network architectures—Feedforward Neural Network (FFNN), Gated Recurrent Unit (GRU), and Bidirectional Long Short-Term Memory (Bi-LSTM)—for the task of multi-class classification in e-commerce product categorization. Each architecture leverages word embeddings to process textual data, and their performances are compared using three optimizers: Stochastic Gradient Descent (SGD), Adaptive Moment Estimation (Adam), and Adaptive Gradient Algorithm (Adagrad). This paper outlines the architecture of each model and evaluates their effectiveness based on accuracy and classification performance. Code is available at https://github.com/MbeleckBerle/ecommerce_product_classification

1 Introduction

E-commerce platforms and online retail businesses continuously face the challenge of efficiently tagging new products to appropriate categories. With millions of products being added daily, manually tagging each product becomes an impractical and costly task. Accurate product categorization is vital, as it improves customer search experiences and optimizes business operations.

In this project, we address this challenge by building predictive models that categorize products based on their textual descriptions. Specifically, the goal is to create a system that can accurately classify products into predefined categories using state-of-the-art machine learning and deep learning techniques. Product categorization is formulated as a supervised classification problem where the product descriptions serve as the input features, and the product categories are the target labels. By employing neural network models, we aim to develop a system that achieves high classification accuracy and precision, ultimately reducing manual effort and operational costs.

2 Methodology

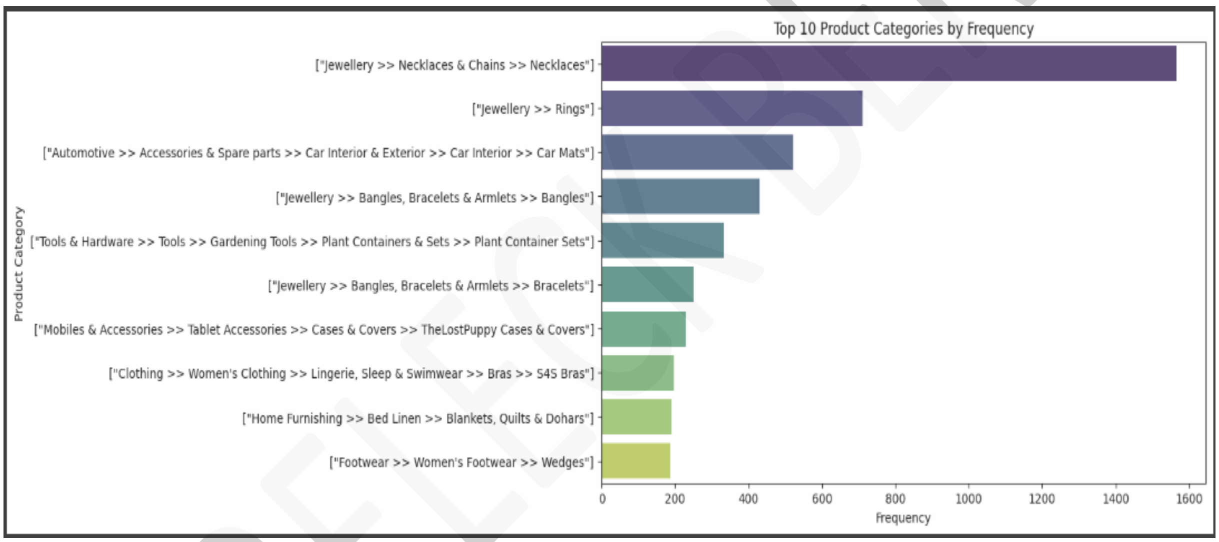
The dataset used in this study consists of product descriptions categorized into multiple classes. It provides comprehensive product details from Flipkart, one of India's leading e-commerce platforms, for the year 2017. it offers valuable insights into the e-commerce landscape of the time, capturing a wide array of product information access various categories
sample source: https://data.world/promptcloud/product-details-on-flipkart-com

updated dataset: https://app.datastock.shop/?site%5C_name=Flipkart%5C_Product%5C_Listings (paid)

Key Features:

- **Product Information:** Detailed descriptions, specifications, and features for thousands of products.
- **Pricing Data:** Includes original prices, discounts, and final sale prices to understand pricing strategies.
- Category Coverage: A wide range of product categories, including electronics, fashion, home appliances, books, and more.
- **Customer Reviews and Ratings:** Aggregated customer feedback to assess product popularity and customer satisfaction.
- **Seller Information:** Data about various sellers, providing insights into market competition and seller performance.

product_url	product_name	product_category_tree	pid	retail_price	discounted_price	
http://www.flipkart.com/alisha-solid-women-s-c...	Alisha Solid Women's Cycling Shorts	["Clothing >> Women's Clothing >> Lingerie, Sl...	SRTEH2FF9KEDEFGF	999.0	379.0	["h
ttp://www.flipkart.com/fabhomedecor-fabric-do...	FabHomeDecor Fabric Double Sofa Bed	["Furniture >> Living Room Furniture >> Sofa B...	SBEEH3QGU7MFYJFY	32157.0	22646.0	["h
http://www.flipkart.com/aw-bellies/p/itmeh4grg...	AW Bellies	["Footwear >> Women's Footwear >> Ballerinas >...	SHOEH4GRSUBJGZXE	999.0	499.0	["h
http://www.flipkart.com/alisha-solid-women-s-c...	Alisha Solid Women's Cycling Shorts	["Clothing >> Women's Clothing >> Lingerie, Sl...	SRTEH2F6HUZMQ6SJ	699.0	267.0	["h
http://www.flipkart.com/sicons-all-purpose-arn...	Sicons All Purpose Arnica Dog Shampoo	["Pet Supplies >> Grooming >> Skin & Coat Care...	PSOEH3ZYDMSYARJ5	220.0	210.0	



Top 10 categories of the dataset

Data Cleaning and Preprocessing

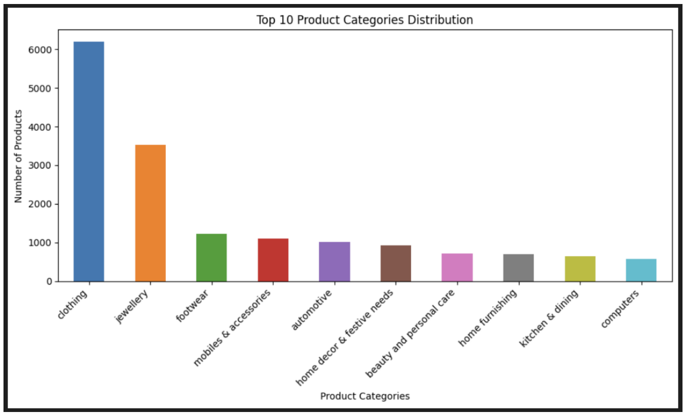
The e-commerce data set has 15 attributes; out of these attributes, only extract the following are used for further analysis

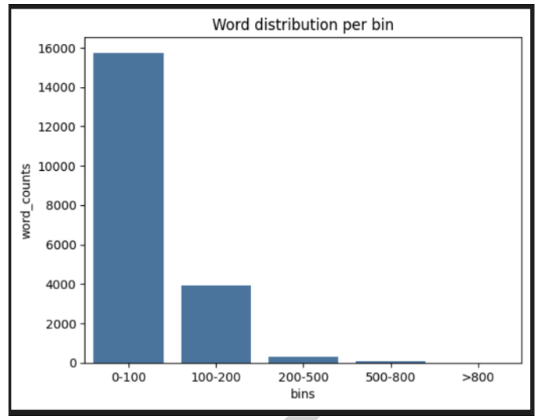
- description
- product_category_tree

The remaining columns are not important for the text classification task. After dropping irrelevant features, the following is performed

- removed punctuations
- removed whitespace between terms
- convert to lowercase
- remove leading whitespace
- Replace numbers with 'numbr'
- selected only the head label from the labels

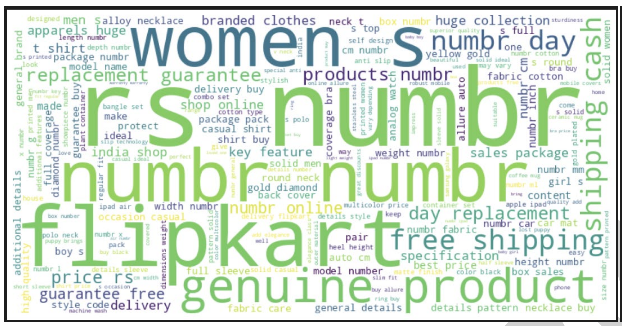
The labels are cleaned, and the main category is kept as the label for simplicity

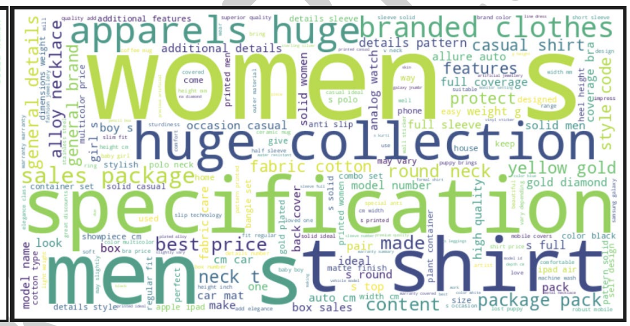




Visualizing the number of words in descriptions. Most of the descriptions have fewer than 200 words. And more than 80% contains fewer than 100 words.

Word cloud after preprocessing





Word cloud after removing domain specific stop words

The text data was tokenized and converted into sequences of numerical values based on a predefined vocabulary size. Each of the three models was trained with three different optimizers (SGD, Adam, Adagrad), and their performance was evaluated on a multi-class classification task. The following subsections detail the architecture of each neural network model used.

3 Model Architectures

Three models are built with the following optimizers

- SGD
- Adam
- Adagrad

3.1. Feedforward Neural Network (FFNN)

The FFNN is a simple, fully connected network that processes text data using an embedding layer followed by dense layers. The architecture includes:

- Embedding Layer: Maps input words to dense vectors of fixed size (128 dimensions), transforming the vocabulary into a numerical format for the neural network.
- Flatten Layer: Converts the 2D embedding output into a 1D vector.
- Dense Layers: A hidden layer with 128 units using ReLU activation, followed by an output layer with softmax activation for multi-class classification.

This network captures word-level features but lacks the ability to capture sequential information in the product descriptions.

3.2. Gated Recurrent Unit (GRU)

The GRU model is a type of recurrent neural network (RNN) designed to capture sequential patterns in text data. The GRU layer handles long-term dependencies in the text without the vanishing gradient problem. The architecture consists of:

- Embedding Layer: Converts input text to dense word embeddings (128 dimensions).
- GRU Layer: A GRU layer with 128 units that captures temporal dependencies in the input sequences.
- Dense Layers: A fully connected layer with 128 units (ReLU activation) and an output softmax layer for multi-class classification.

This architecture is well-suited for handling sequential data and offers better context understanding than FFNN.

3.3. Bidirectional Long Short-Term Memory (Bi-LSTM)

The Bi-LSTM is an advanced variant of RNN, where the network processes the input in both forward and backward directions. This allows the model to capture more contextual information. The architecture is as follows:

- Embedding Layer: Maps input words to dense vectors (128 dimensions).
- Bidirectional LSTM Layer: A Bidirectional LSTM layer with 128 units, which processes sequences from both directions, capturing future and past context.
- Dense Layers: A dense layer with 128 units (ReLU activation) followed by a softmax output layer for classification.

Bi-LSTM can handle long-term dependencies more effectively than GRU and FFNN due to its bidirectional nature.

4 Results

In this study, we evaluated the performance of three distinct neural network architectures—Feedforward Neural Network (FFNN), Gated Recurrent Unit (GRU), and Bidirectional Long Short-Term Memory (BI-LSTM)—across three different optimizers: Stochastic Gradient Descent (SGD), Adam, and Adagrad. The primary metric for assessment was model accuracy, which is indicative of each model's effectiveness in classifying the test data.

Model Performance

The accuracy results for each model and optimizer combination are summarized in Table 1.

Model	SGD	Adam	Adagrad
FFNN	60.8	92.8	44.2
GRU	45.6	92.4	31.5
BI-LSTM	46.5	92.0	31.5

From the results, it is evident that the Adam optimizer consistently outperformed the other optimizers across all model architectures. The FFNN achieved the highest accuracy of **92.8%** with the Adam optimizer, significantly surpassing the SGD and Adagrad results, which recorded **60.8%** and **44.2%** accuracy, respectively.

Similarly, both GRU and BI-LSTM models demonstrated comparable performance, with the Adam optimizer achieving accuracies of **92.4%** and **92.0%** respectively, while the accuracy with SGD was notably lower at **45.6%** and **46.5%** for GRU and BI-LSTM, respectively. Adagrad exhibited the lowest accuracy among all optimizers across all models, with the GRU and BI-LSTM yielding **31.5%** accuracy.

5 Interpretation of Results

The superior performance of the Adam optimizer can be attributed to its adaptive learning rate and momentum capabilities, allowing for better convergence during training. This is particularly crucial in deep learning models where the loss landscape can be complex. Conversely, the lower accuracy achieved with SGD may indicate its sensitivity to the learning rate and the potential for slower convergence.

In conclusion, these results highlight the importance of selecting the appropriate optimizer in training neural network models. The findings suggest that for the given task, particularly when using FFNNs, GRUs, and BI-LSTMs, the Adam optimizer is the most effective choice, facilitating higher accuracy compared to SGD and Adagrad.

6 Conclusion

This study investigated the impact of different neural network architectures and optimization algorithms on model accuracy in a classification task. By evaluating Feedforward Neural Networks (FFNN), Gated Recurrent Units (GRU), and Bidirectional Long Short-Term Memory (BI-LSTM) networks using Stochastic Gradient Descent (SGD), Adam, and Adagrad optimizers, we gained valuable insights into the efficacy of these models.

The results demonstrated that the Adam optimizer consistently outperformed both SGD and Adagrad across all architectures, achieving the highest accuracy of **92.8%** with the FFNN model. The GRU and BI-LSTM models also benefitted from the Adam optimizer, yielding accuracies of **92.4%** and **92.0%**, respectively. In contrast, the Adagrad optimizer resulted in the lowest accuracies for all models, while SGD produced moderate performance, particularly in the case of FFNN.

These findings underscore the critical role of optimizer selection in training neural networks. The superior performance of the Adam optimizer highlights its effectiveness in handling complex loss landscapes, making it a preferred choice for deep learning applications. Overall, this study contributes to the understanding of model training dynamics and provides a foundation for further exploration of optimization strategies in neural networks.

Future research may focus on exploring advanced optimization techniques and regularization methods to enhance model performance further. Additionally, investigating the effects of hyperparameter tuning and feature engineering could provide deeper insights into improving classification accuracy in similar tasks.

7 References

- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780
- Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. EMNLP.
- https://data.world/promptcloud/product-details-on-flipkart-com
- https://app.datastock.shop/?site%5C_name=Flipkart%5C_Product%5C_Listings